

Assignment - 1

Q.1 Load the contents of the memory location 2200H and 2201H into registers. Add these registers and ~~some~~ store the result in memory locations 2202H and 2302H.

Soln:

OPCODE	PSEUDO CODE
21 00 22	lxi h, 2200h ; $\text{HL} = 2200\text{h}$
46	mov b, m ; $b = 1^{\text{st}}$ number
23	inx h ; $HL = 2201\text{h}$
7E	mov a, m ; $a = 2^{\text{nd}}$ number
23	inx h ; $HL = 2202\text{h}$
OE 00	mvi c, 00h ; $c = 00\text{h}$
80	add b ; $a = a + b$
D2 0e 08	jnc skip ; jump if result \rightarrow no carry
0C	inx c ; else $c = c + 1$
77	skip : mov m, a ; $[D202\text{h}] = \text{sum}$
79	mov a, c ; $a = c$, which is the carry
23	inx h ; $HL = 2203\text{h}$
77	mov m, a ; $2203\text{h} = \text{carry}$
76	hlt ; program ends.

R. Jaskaran
29.05.23

Memory Address	OPCODE
2000	21
2001	00
2002	22
2003	56
2004	21
2005	00
2006	25
2007	0E
2008	00
2009	3E
200A	46
200B	23
200C	80
200D	D2
200E	12
200F	08
2010	0C
2011	15
2012	C2
2013	06
2014	08
2015	032
2016	00
2017	23
2018	79
2019	32
201A	01
201B	23
201C	76

Q2. Find the sum of N numbers stored in consecutive locations starting from 2500H. The value of N is stored in 2200H. Store the result in locations 2300H and 2301H.

Soln.

OPCODE	ASSEMBLY CODE
21 00 22	lxi h, 2200H ; HL = 2200H
56	mov d, m ; d = [2202] (count)
21 00 25	lxi h, 2500H ; 2200H HL = 2500H (starting address)
OE 00	mvri c, 00H ; C = 00H (carry)
3E 00	mvri a, 00H ; A = 00H
46	back : mov b, m ; B = 1st number
23	inx h
80	add b ; A = A + B (sum of first two no.)
D2 12 08	jnc skip
0C	inc C ; if it produces a carry then
15	skip : dcr d ; d = d - 1 (numbers to add)
C2 0B 08	jnz back
32 00 23	sta 2300H ; [2300H] = result

R. Dasgupta
24-05-23

Memory Address

opcode

Memory Address	opcode
2000	21
2001	00
2002	25
2003	7E
2004	E6
2005	47
2006	7E
2007	E6
2008	f0
2009	0F
200A	0F
200B	0F
200C	0F
200D	0F
200E	80
200F	32
2010	50
2011	25
2012	76
2013	30 11 20
2014	20
	31
	C5 00 50
	22 00 25

new code
ex. 00
00 00

Q.3) Find the sum of least significant 4 bit and most significant 4 bits of a byte stored in memory location 2500H. store the result in 2550H.

Assembly Code

Comments

ldi h, 2500H ; HL = 2500H

mov a, m ; a = [2500H]

andi 0fh ; (AND op) masking → keep → lower 4 bits

mov b, a ; b = a (lower 4 bits)

mov a, m ; a = [2500H]

andi f0h ; masking → keep → higher 4 bits

rrc ; right rotating it 4 times

rrc

rrc

rrc ; now e.g., 30 will become 03

add b ; a = a + b

sta 2550h ; [2550] = result

hlt ; program ends

R. Dasika
29-05-23

Memory Address	Opcode
2000	21
2001	00
2002	25
2003	7E
2004	06
2005	00
2006	16
2007	08
2008	0F
2009	02
200A	0d
200B	08
200C	04
200D	15
200E	C2
200F	08
2010	78
2011	32
2012	10
2013	26
2014	3E
2015	08
2016	90
2017	32
2018	11
2019	25
201A	76
201B	

Q4) Write a program to count the '1's and '0's of a byte stored in 2500H. Store the result in 2610H and 2511H respectively.

Assembly code

Comments

ldi h, 2500H	;	HL = 2500H
mov a, m	;	a = [2500H]
mvf b, 00h	;	stores count of 1.
mvf d, 08h	;	d=number of bits (8)
back: rrc	;	right - rotate
jnc zero	;	jump → if carry == 0
inx b	;	b = b + 1
zero: dcr d	;	d = d - 1
jnz back	;	jump → d != 0
mov a, b	;	a = count of 1
sta 2610h	;	[2610] = count of 1
mvf a, 08h	;	a = 08h
sub b	;	a = a - b (count of zeros)
sta 2511h	;	[2511] = count of 0
hlt	;	program ends

R. Jaskar 24-05-23

Memory Address	Assembly Code	Opcode
2000		3A
2001		00
2002		25
2003		0E
2004		00
2005	HL = 2000H	47
2006	A = [2000]	3A
2007	to from = A	02
2008	pushes of p	55
2009	Stops trin	80
200A	0 == 2000	32
200B	I + d = d	00
200C	I - b = b	26
200D	a = !b ← qmvi	3A
200E	I to from = A	01
200F	I to from = [0000]	25
2010	d > 0 = 0	47
2011	d = 0 = 0	3A
2012	0 to from = [1111]	03
2013	ebus memory	025
2014		88
2015	20.10.2014	D2
2016		19
2017		08
2018		0C
2019		32
201A		01

Q5> Write a program to sum two 16-bits binary numbers.

Assembly code	Comments
lda 2500h	a = lower byte of 1st number
mvi c, 00h	c = 00h
mov b, a	b = a
lda 2502h	a = lower byte of 2nd number
add b	a = a+b
sta 2600h	lower byte of result → [2600h]
lda 2500h	a = higher byte of 1st number
mov b, a	b = a
lda 2503h	a = higher byte of 2nd number
adc b	a = a+b + previous carry
jnc skip	skip → if carry != 1
inx c	c = c+1
skip : sta 2601h	[2601h] = higher byte → result
mov a, c	a = c
sta 2602h	[2602h] = higher byte of result
hlt	program ends

Memory Address	Opcode
201B	26
201C	79
201D	32
201E	02
201F	26
2020	76

Memory Address	Opcode
2000	3A
2001	50
2002	20
2008	E6
2004	0F
2005	47
2006	3A
2007	51
2008	20
2009	E6
200A	f0
200B	80
200C	16
200D	C4
200E	0F
200F	15
2010	C2
2011	0e
2012	08
2013	32
2014	52
2015	20
2016	3A
2017	50
2018	20
2019	20
201A	16
201B	E6
201C	47
201D	3A
201E	51
201F	20
2020	E6
2021	0f
2022	80
2023	16
2024	04
2025	0F
2026	15
2027	C2
2028	24
2029	08
203A	32
203B	53
	20
	76

Assignment-2

1. Two numbers MN_H and KL_H are stored in 2050_H and 2051_H respectively. Write a program to assemble them as NK_H and LM_H store them in 2052_H and 2053_H .

Assembly codeComments

```

lde 2050h ; a=[2050h]
ani ofh ; a=ON
mov b,a ; b=a
lde 2051h ; a=[2051h]
ani foh ; a=KO
add b ; a=a+b (KN)
mvi d,04H ; d=04H
back1: rrc ; right rotate
      dcr d ; d=d-1
      jnz back1 ; jump if d!=0
sta 2052h ; [2052h]=NK
lde 2050h ; a=[2050h]
ani foh ; a=NO
mov b,a ; b=a
lde 2051h ; a=[2051h]
ani ofh ; a=OL
add b ; a=a+b (ML)
mvi d,04H ; d=04H
back2: rrc ; right rotate
      dcr d ; d=d-1
      jnz back2 ; jump if d!=0
MERIT® sta 2053h ; [2053h]=Nk
hlt ; program ends

```

R. Dandekar
29-05-23

Memory Addressopcode

Memory Address	Address	opcode
2000		21
2001		00
2002		00
2003		11
2004		00
2005		00
2006		3A
2007		50
2008	$a = [soe10]$	20
2009	$DN = 0$	C6
200A		00
200B	$a = d$	CA
200C	$a = [soe11]$	1d
200D	$DN = 0$	08
200E	$(H) d + a = 0$	SF
200F	$HPO = b$	3A
2010	$a = D$	51
2011	stator Werr	20
2012	$i - b = b$	C6
2013	$d = b$	00
2014	$a = b$	CA
2015	$[soe23] = NK$	1D
2016	$a = [soe20]$	08
2017		4F
2018	$DN = 0$	19
2019	$a = D$	0D
201A		C2
201B	$a = [soe11]$	18
201C	$D = 0$	08
201D	$D = 0$	7D
201E	$(H) d + a = 0$	32
201F	$HPO = b$	52
2020	stator Tneir	20
2021	$i - b = b$	7C
2022	$a = b$	32
2023	$a = b$	53
2024	$[soe23] = NK$	20

2. Two numbers A & B are stored in 2050H and 2051H respectively. Write a program to perform $A \times B$ and store the result in 2052_H and 2053_H .

Assembly code
Comments

```

lxi h, 0000h ; HL = 0000H
lxi d, 0000h ; DE = 0000H
lda 2050h ; A = [2050h]
adi 00h ; checking 1st operand == 0
jz store ; 0 * something = 0, so store 0
mov e, a ; PE = 00XXH
lda 2051H ; A = [2051h]
adi 00h ; checking second operand == 0
jz store ; something * 0 = 0, so store 0
mov c, a ; C = A
back: add d, d ; HL = HL + DE
dec c ; C = C - 1
jnz back ; jump if C != 0
store: mov a, l ; A = L
sta 2052h ; [2052h] = lower byte of result
mov a, h ; A = H
sta 2053h ; [2053h] = higher byte of result
hlt ; program ends.

```

 R. Darshan
24-05-23

Memory Address	Opcode	Memory Address	Opcode
2000	3A	2024	3A
2001	4F	2025	4F
2002	20	2026	20
2003	3D	2027	3D
2004	47	2028	21
2005	21	2029	50
2006	50	202A	20
2007	20	202B	85
2008	3A	202C	D2
2009	4F	202D	36
200A	20	202E	08
200B	H00003D	202F	08
200C	H00004F	2030	24
200D	7E	2031	6F
200E	23	2032	7E
200F	BE	2033	32
2010	DA	2034	61
2011	1C	2035	0120
2012	08	2036	1676
2013	5E		
2014	77		
2015	2B		
2016	7B		
2017	77		
2018	23		
2019	0D		
201A	C2		
201B	08		
201C	05		
201D	C2		
201E	05		
201F	08		
2020	3A		
2021	50		
2022	20		
2023	32		
	60		
	20		

3. N numbers are stored in consecutive m/m location starting from 2050_{16} . The value N is stored in $204F_{16}$.

- i) Find the maximum among the N numbers
- ii) Find the minimum among the numbers
- iii) Sort the N numbers in ascending order
- iv) Sort the N numbers in descending order

<u>Opcode</u>	<u>Assembly code</u>	<u>Comments</u>
8A 4f 20	ld a 204Fh ;	a=N (count)
3D	dec a ;	a=a-1
47	mov b,a ;	b=a (outer loop count)
21 50 20	back2: lxi h, 2050h ;	HL=2050h
3A 4f 20	ld a 204Fh ;	a=N (count)
3D	dec a ;	a=a-1
4F	mov c,a ;	c=a (inner loop count)
7E	back1: mov a,m ;	d=1st number
23	inx h ;	HL=2051H
BE	cmp m ;	a-b
DA 1c 08	jcs skip ;	i.e. a < b
CA 1c 08	jz skip ;	i.e. a == b
5E	mov e,m ;	e=2nd number
77	mov en,a ;	[2051] = 1st number
2B	dex h ;	HL=HL-1
7B	mov a,e ;	a=2nd number
77	mov m,a ;	[2050h] = 2nd number
23	inx h ;	HL=2051h

DATEPAGE NO.

AS

TP

OS

0002

19

2000

EXPT. NO.

<u>Opcode</u>	<u>Assembly Code</u>	<u>Comments</u>
0D	skip: dec C ;	C = C - 1
C2 0D 08	jnz back1 ;	if C1 = 0
05	dec b ;	b = b - 1
C2 05 08	jnz back2 ;	if b ≠ 0
3A 50 20	lda 2050h ; a = 1st number (smallest)	
32 60 20	sta 2060h ; [2060h] = a	
3A 4F 20	lda 204Fh ; a = N (count)	
3D	dec a ;	a = a - 1
21 50 20	lxi h, 2050h ; HL = 2050h	
85	add l ;	a = a + l
D2 36 08	jnc store ; if no carry	
24	inx h ;	h = h + 1
6F	store: mov b, a ; b = a	
7E	mov a, m ; a = largest number	
32 61 20	sta 2061h ; [2061h] = last number	
76	hlt ; program ends	(largest)

✓ Raju Darshan

24-05-23

<u>Memory Address</u>	<u>Opcode</u>
2000	3A
2001	4F
2002	20
2003	3D
2004	47
2005	21
2006	50
2007	3A
2008	4F
2009	20
200A	3D
200B	4F
200C	7E
200D	2B
200E	BE
200F	02
2010	1C
2011	08
2012	CA
2013	1C
2014	08
2015	SE
2016	77
2017	7B
2018	2B
2019	77
201A	23
201B	0D
201C	C2
201D	0d
201E	08
201F	05
2020	C2
2021	05
2022	08

Assembly

<u>Opcode</u>	<u>Assembly code</u>	<u>Comments</u>
3A 4f 20	ld a 204Fh ;	a=N (count)
3D	dcr a ;	a=a-1
47	mov b,a ;	b=a (outer loop count)
21 50 20	back2: lxi h, 2050h ;	HL=2050h
3A 4f 20	ld a 204Fh ;	a=N (count)
3D	dcr a ;	a=a-1
4F	mov c,a ;	c=a (inner loop count)
7E	back1: mov a,m ;	d=1st number
23	inx h ; HL=2051	HL=2051
BE	comp m ;	a-b
D2 1C 08	jnc skip ;	i.e., a>b
CA 1C 08	jz skip ;	i.e., a==b
5E	mov e,m ;	e = 2nd number
77	mov m,a ;	[2050] = 1st number
2B	dcr h ;	HL=2050
7B	mov a,e ;	a = 2nd number
77	mov m,a ;	[2050h] = 2nd number
23	inx h ;	HL=2051
00	skip: dcr c ;	c=c-1
C2 0d 08	jnz back1 ;	jump if c!=0
05	dcr b ;	b=b-1
C2 05 08	jnz back2 ;	jump if b!=0
76	hlt	; program ends

R. Jaskaran
24 - 05 - 23

<u>Memory</u>	<u>Address</u>	<u>Opcode</u>	<u>Memory</u>	<u>Address</u>	<u>Opcode</u>
2000		3A	2024		50
2001		4f	2025		20
2002		20	2026		11
2003		06	2027		06
2004		00	2028		22
2005		4f	2029		7E
2006		21	202A		0F
2007		50	202B		D2
2008		20	202C		43
2009		11	202D		08
200A		00	202E		04
200B		21	202F		FB
200C		7E	2030		0F
200D		OF	2031		77
200E		DA	2032		23
200F		1d	2033		EB
2010		08	2034		23
2011		04	2035		0P
2012		EB	2036		C2
2013		77	2037		32
2014		23	2038		08
2015		EB	2039		78
2016		23	203A		32
2017		0D	203B		01
2018		C2	203C		23
2019		0C	203D		76
201A		08	203E		88
201B		78	203F		00
201C		32	2040		00
201D		00	2041		23
201E		23	2042		3A
201F		3A	2043		4f
2020		4f	2044		20
2021		20	2045		06
2022		06	2046		00
2023		00	2047		4F
			2048		21

4. N numbers are stored in consecutive m/m locations starting from 2050_{16} . The value of N is stored in $204F_{16}$. Write a program to copy the even no. and odd no starting from 2100_{16} and 2200_{16} , respectively. Store the total no. of even and odd numbers in 2300_{16} and 2301_{16} respectively.

<u>Opcode</u>	<u>Assembly code</u>	<u>Comments</u>
3A 4f 20	ld a 204fh ;	a=N (count)
06 00	mvi b, 00h ;	b=00h
4F	mov c,a ;	c=a
21 50 20	lx i h, 2050h ;	HL = 2050_{16}
11 00 21	lx i d, 2100h ;	DE = 2100_{16}
7E	backl: mov a,m ;	a=1st number
0F	r8c ;	right rotate \rightarrow last bit \rightarrow carry
DA 1d 08	jc skip1	skip if lsb==1 (odd)
04	inx b ;	b=b+1
EB	xchg ;	HL \leftrightarrow DE
0EOF	r8c ;	left rotate
77	mov m,a ;	$[2100] = 1^{st}$ even no.
23	inx h ;	HL = 2101_{16}
EB	xchg ;	HL \leftrightarrow DE
23	skip1: inx H ;	HL = 2051_{16}
00	dcr c ;	c=c-1
C2 0c 08	jnz backl	
78	mov a,b ;	a=count of even
32 00 23	sta 2300H ;	$[2300H] = \text{count of even}$

DATE

PAGE NO.

EXPT. NO.

OpcodeAssembly codeComments

3A 4f 20	ld a 204fh ;	a=N (count)
06 00	mv r b, 00h ;	b=00h
4F	mov c, a ;	c=a
21 50 20	lx i h, 2050h ;	HL = 2050h
11 00 22	lx i d, 2200h ;	DE = 2200h
7E	back2: mv a, m ;	a = 1st number
0F	rrc ;	right rotate \rightarrow last bit \rightarrow carry
D2 43 08	jnc skip2 ;	skip if lsb==0 (even)
04	ins b ;	b=b+1
EB	xchg ;	DE \leftarrow HL
0F	rcl ;	left rotate \rightarrow original num
77	mv m, a ;	[2200] = 1st odd number
23	inx h ;	HL = 2201H
EB	xchg ;	HL \leftrightarrow DE
23	skip2: inx h ;	HL = 2051h
0D	dcr c ;	c=c-1
C2 32 08	jnz back2	
78	mv a, b ;	a = count of odd
32 01 23	sta 2301h ;	[2301] = count of odd
76	hlt ;	program ends

R. Dasgupta
29-05-23

<u>Memory</u>	<u>Address</u>	<u>opcode</u>
2000		3F
2001		4f
2002		32
2003	→ 600 1014	4D
2004	→ A11005 0000	20
2005	→ 400 0000	3A
2006	→ 100 0000	4f
2007	→ 100 0000	20
2008	→ 100 0000	4E
2009	→ 1000 1014	01
200A	→ 1000 1014	21
200B	→ 1000 1014	50
200C	→ 1000 1014	20
200D	→ 1000 1014	3A
200E	→ 1000 1014	4E
200F	→ 1000 1014	20
2010	→ DE → HF	47
2011	→ F1	7E
2012	→ [0005]	C2
2013	→ 14	90
2014	→ DE ← HF	08
2015	→ 14	7A
2016	→ 14	32
2017	→ 14	4d
2018	→ 14	20
2019	→ 14	23
201A	→ [0005]	14
201B	→ [0005]	0D
201C	→ 14	C2
201D	→ 14	12
201E	→ 14	08
201F	→ 14	76
2020	→ 14	
2021	→ 14	
2022	→ 14	
2023	→ 14	
2024	→ 14	

5. N numbers are stored in consecutive m/m location starting from 2050_H . The value of N is stored in $204F_H$. Write a program to test whether a number stored in $204F_H$ is present in the list. If present, store its position in the list at $204D_H$; otherwise store FF_H .

<u>Opcode</u>	<u>Assembly code</u>	<u>Comments</u>
3E ff	mvi a, FFh ;	a=FFh (assuming not present)
32 40 20	sta 2040h ;	[2040h]=44h
3A 4f 20	lda 204fh ;	a=count
4F	mov c,a ;	c=count
16 01	mvi d, 01h ;	d=01h (index)
21 50 20	inx h, 2050h ;	HL=2050h
3A 4e 20	lda 204eh ;	a=target number
47	mov b,a ;	d=a (target number)
7E	back: mov a,m ;	a=1st number
90	sub b,	a=a-b (target no.)
C2 1b 08	jnz skip ;	jump if result != 0
7A	mov a,d ;	a=current index
32 4d 20	sta 204Dh ;	[204Dh]=index result
23	skip: inx h ;	HL=2051h
14	inx d ;	d=d+1 (index)
0D	dec c ;	c=c-1 (count)
C2 12 08	jnz back ;	if c!=0
76	ret ;	program ends.

<u>Memory</u>	<u>Address</u>	<u>Opcode</u>
	2000	3A
	2001	00
	2002	25
	2003	47
	2004	21
	2005	01
	2006	25
	2007	11
	2008	00
	2009	26
	200A	7E
	200B	4F
	200C	0F
	200D	D2
	200E	02
	200F	08
	2010	FB
	2011	10
	2012	71
	2013	EB
	2014	01
	2015	02
	2016	07
	2017	07
	2018	DA
	2019	10
	201A	08
	201B	23
	201C	05
	201D	C2
	201E	0a
	201F	08
	2020	76
	2021	

R. Daskar

24-05-23

Assignment- 3

1. A set of N data bytes is stored in m/m locations starting from 2501_{16} . The value of N is stored in 2500_{16} . Write a program to store these data bytes from m/m location 2600_{16} if D_0 or D_7 is 1; otherwise reject the data byte.

<u>Opcode</u>	<u>Assembly Code</u>	<u>Comments</u>
3A 00 25	lda 2500h ;	a = [2500h] (count)
47	mov b,a ;	b = a count
21 01 25	lxr h, 2501h ;	HL = 2501h
11 00 26	lxr d, 2600h ;	DE = 2600h
7E	back: mov o,m ;	a = 1st number
4F	mov c,a ;	c = b
0F	rxc	right rotate
D2 17 08	jnc check2 ;	lcb == 0
EB	store:xchg ;	DE \leftrightarrow HL
71	mov m,c ;	[2600] = result
EB	xchg	DE \leftrightarrow HL
13	inx d	DE = 2601h
C3 1C 08	jmp loop	unconditional jump
07	check2: rlc ;	msb \rightarrow carry flag and lsb
07	rlc ;	msb \rightarrow carry flag of lsb
DA 10 08	jc store ; msb == 1	
23	loop: inx h	HL = 2502h
05	dec b	b = b - 1
C2 0A 08	jnz back ;	if b != 0
MERIT® 76	hit	program ends

<u>Memory</u>	<u>Address</u>	<u>Opcode</u>	<u>Memory</u>	<u>Address</u>	<u>Opcode</u>
2000		3E	2023		17
2001		06	2024		08
2002		32	2025		23
2003		27	2026		05
2004		3A	2027		C2
2005		ff	2028		11
2006		21	2029		08
2007		47	203A		3A
2008		0E	202B		00
2009		00	202C		27
200A		21	202D		57
200B		00	202E		21
200C		21	202F		00
200D		00	2030		6E
200E		22	2031		00
200F		11	2032		3F
2010		00	2033		00
2011		26	2034		46
2012		7E	2035		80
2013		4F	2036		02
2014		0F	2037		40
2015		D2	2038		08
2016		25	2039		0C
2017		08	203A		23
2018		EB	203B		15
2019		71	203C		22
201A		13	203D		39
201B		3A	203E		08
201C		00	203F		32
201D		27	2040		32
201E		1C3	2041		01
201F		12A	2042		25
2020		08	2043		76
2021		07			
2022		07			
		DA			

2. There are N data bytes stored from memory location 2200_4 . The value of N is stored in $21FF_4$. Write an 8085 program to find the sum of integers whose LSB and MSB are 1. Store the result in 2500_4 and 2501_4 .

<u>Opcode</u>	<u>Assembly Code</u>	<u>Comments</u>
3E 00	MVI A, 00h ;	a=00h
32 00 27	STA 2700h ;	[2700h] = 2700
3A ff 21	LDI A, 21ffh ;	a=[21ffh]
47	MOV B, A ;	b=a
0E 00	MVI C, 00h ;	C=00h
21 00 22	LDI H, 2200h ;	HL=2200h
11 00 26	LDI D, 2600h ;	DE=2600h
7E	back : MOV A, M	a = 1st number
4F	Mov C, A ;	c=a
OF	RCR ;	LSB → carry flag
D2 25 08	JNC check2 ;	LSB == 0
EB	Store: XCHG ;	DE ↔ HL
71	Mov M, C ;	[2600h] = result
EB	XCHG ;	DE ↔ HL
13	INX D	- DE += 1
3A 00 27	LDI A, 2700h ;	a=[2700]
3C	INR A	- a=a+1
32 00 27	STA 2700h ;	[2700] = result
C3 2A 08	JMP Loop ;	unconditional jump

DATE

PAGE NO.

EXPT. NO.

OpcodeAssembly codeComments

07	check2: rlc	; right rotate
07	rlc	"
DA 17 08	jc store	; msb == 1
23	loop : inx h ;	HL = 2502
05	dcr b	; b = b - 1
C2 11 08	jnz back ;	if b != 0
3A 00 27	lda 2708h ;	a = [2708h]
57	mov d, a	; d = a
21 00 26	lxi h, 2600h	HL = 2600h
0E 00	mvi c, 00h	; c = 00h
3E .00	mvi a, 00h	; a = 00h
46	backnew: mov b, m	; b = 1 st number
80	add b	; a = a + b
D2 40 08	jnc skipnew	; if no carry
0C	inx c	; c = c + 1
23	skipnew: inx h	; HL = HL + 1
15	dcr d	; d = d - 1
C2 3A 08	jnc backnew	
32 60 25	sta 2508h	; [2500h] = sum
79	mov a, c	; a = c
32 01 25	sta 2501h	; [2501h] = carry
76	hlt	; program ends

R. Naskar

24-05-23

Memory addressOpcodeMemory Address

Memory address	Opcode	Memory Address	Opcode
2000	3E	2021	21
2001	00	2022	79
2002	32	2023	32
2003	00	2024	01
2004	21	2025	21
2005	3E	2026	32
2006	01	2027	20
2007	32	2028	15
2008	01	2029	C2
2009	21	202A	18
200A	3A	202B	08
200B	60	202C	76
200C	20	202D	60
200D	C6	202E	20
200E	00	202F	32
200F	CA	2030	50
2010	31	2031	20
2011	08	2032	E3
2012	D6	2033	30
2013	01	2034	08
2014	CA		02
2015	31		
2016	08		30
2017	57		82
2018	3A		21
2019	01		
201A	21		
201B	47		
201C	3A		
201D	00		
201E	21		
201F	80		
2020	4F		
	78		
	32		
	00		
	21		

3. Write an 8085 program to generate N^{th} fibonacci number using function and store it in 2050_{H} . The value of N (8-bits) is stored in memory 2060_{H} .

<u>Opcode</u>	<u>Assembly Code</u>	<u>Comments</u>
3E 00	mvi a, 00h ;	a=00h
32 00 21	sta 2100h ;	[2100h]=00
3E 01	mvi a, 01h ;	a=01h
32 01 21	sta 2101h ;	[2101h]=01h
3A 60 20	lda 2060h ;	a=[2060h]
C6 00	adi 00h ;	a=a+00h;
CA 31 08	jz storen ;	store n as result if $n==0$
D6 01	sui 01h ;	a=a-01h
CA 31 08	jz storen ;	store n as result if $n==1$
S7	mov d,a ;	d=a (count)
3A 01 21	back: lda 2101h ;	a=[2101h]
47	mov b,a ;	b=a (i-1)
3A 00 21	lda 2100h ;	a=[2100h]
80	add b ;	a=a+b
4F	✓ mov c,a ;	c=result
78	mov a,b ;	a=b
32 00 21	sta 2100h ;	[2100h]=(i-2)
79	mov a,c ;	a=c
32 01 21	sta 2101h ;	[2101h]=i-1
32 50 20	sta 2050h ;	[2050h]=result
15	dec d ;	d=d-1



DATE

PAGE NO.

18

51

EXPT. NO.

<u>Opcode</u>	<u>Assembly Code</u>	<u>Comments</u>
C2 18 08	Jnz back	
76	end :hlt	
3A 60 20	Storen : lda 2060h ; a = [2060h]	
32 50 20	sta 2050h ; [2050] = count	
C3 30 08	jmp end	

R. Jaskaran

29-05-23

Me

<u>Memory Address</u>	<u>Opcode</u>
2000	31
2001	1e
2002	20
2003	47
2004	3A
2005	1f
2006	20
2007	4F
2008	21
2009	00
200A	21
200B	78
200C	3D
200D	6F
200E	54
2010	5D
2011	78
2012	91
2013	83
2014	D2
2015	17
2016	08
2017	14
2018	5F
2019	7E
201A	EB
201B	77
201C	2B
201D	EB
201E	2B
201F	05
2020	C2
2021	18
2022	08
2023	76
2024	

Q4. Write a program to transfer a block of bytes of size N from location 1 to location 2 (location 2) location 1) when the size of overlap between two locations is defined by M. The values of M and N are stored in 201EH and 201FH, respectively.

<u>Opcode</u>	<u>Assembly code</u>	<u>Comments</u>
31 1e 20	lda 201eh ;	a=N
47	mov b,a ;	b=a
3A 1f 20	lda 201fh ;	a=M
4F	mov c,a ;	c=a(M)
21 00 21	lxr h, 2100h ;	HL=2100
78	mov a,b ;	a=b
3D	dor a ;	a=a-1
6F	mov l,a ;	l=N-1
54	mov d,h ;	D=H
5D	mov e,l ;	E=L
78	mov a,b ;	a=b
91	sub c; ;	a=a-c
83	add e; ;	a=a+e
D2 17 08	jnc skip	
14	inx d;	
SF	skip; mov e,a ;	e=a
7E	back: mov a,m ;	a=last number
EB	xchg	; HL \leftrightarrow DE
77	mov m,a ;	last number \rightarrow last pos

<u>Opcode</u>	<u>Assembly Code</u>	<u>Comments</u>
2B	dcx h ; #	HL=HL-1
EB	xchq ;	DE \leftrightarrow HL
2B	dcx h ;	HL=HL-1
05	dec b ;	b=b-1
C2 18 08	jnz back :	if b!=0
76	hlt ;	program ends

R. DasKan

29-05-23

Memory AddressOpcode

2000

31

2001

FF

2002

20

2003

CD

2004

47

2005

03

2006

AF

2007

47

2008

21

2009

50

200A

20

200B

CD

200C

D0

200D

05

200E

3E

200F

01

2010

06

2011

00

2012

21

2013

54

2014

20

2015

CD

2016

D0

2017

05

2018

11

2019

00

201A

00

201B

CD

201C

BC

201D

03

201E

CD

201F

47

2020

03

2021

11

2022

00

2023

00

2024

CD

2025

BC

2026

03

2027

C3

2028

06

2029

00

202A

20

5) Write a program to flash "BCSE 11" in the address and data fields with a flashing rate of 0.5 seconds.

<u>Opcode</u>	<u>Assembly Code</u>	<u>Comments</u>
91 FF 20	LXI SP, 20FF	Initialize stack pointer
CD 47 03	CALL CLEAR	Clear the display
AF	START XRA A	A is 00 to display characters in address field.
47	MOV B, A	No dot to be displayed in the data field
21 50 20	LXI H, 2050	HL HL = 2050H
CD D0 05	CALL OUTPUT	calling output routine
3E 01	MVI A, 01H	A, 01 initialize ArB for data to be displayed
06 00	MVI B, 00H	B = 00H
21 54 26	LXI H, 2054H	HL = 2054H
CD D0 05	CALL OUTPUT	Display last two characters
11 00 00	LXI D, 0000	
CD BC 03	CALL DELAY	Delay → 0.5 s
CD 47 03	CALL CLEAR	Clear the display
11 00 00	LXI D, 0000	Clear display for 0.5 sec.
CD BC 03	CALL DELAY	
C3 06 20	JMP START	

2050 → OB

2051 → OC

2052 → OS

2053 → OE

2054 → 13

2055 → 13

R. Dasgupta
31-05-23